

### Заключение

В условиях многооператорности и конкуренции на рынке предоставления инфо-телекоммуникационных услуг более рациональным, с точки зрения автора статьи, является использование при проектировании данных сетей критерия максимума прибыли оператора. При использовании данного критерия в процессе проектирования, дополнительно появляется такой варьируемый параметр, как перечень предоставляемых услуг и множество абонентов, которым предоставляются услуги. В результате решаемая оптимизационная задача является более близкой к задачам, решаемым при создании бизнес планов оператор связи.

В статье приведена математическая модель и постановка задачи синтеза топологии и параметрического синтеза мультисервисной сети согласно критерию максимума прибыли оператора связи

### Литература

1. Кучерявый А.Е., Кучерявый Е.А. Иерархические и молекулярные сети связи общего пользования // Электро-связь. 2008. № 2. С. 16-18. 2. MUSE deliverable DTF1.1. Reference Models for a European Multi-service Access Network. January 2006.
2. Зайченко Ю.П. Структурная оптимизация сетей ЭВМ / Зайченко Ю.П., Гонга Ю.В. К.: Техника, 1986. 168 с.

УДК 681.3 (075.8) 378.16

# РЕАЛІЗАЦІЯ ТА ІНТЕГРАЦІЯ ОДНОПАРАМЕТРИЧНОЇ МОДЕЛІ РАША В СИСТЕМУ ДИСТАНЦІЙНОГО НАВЧАННЯ MOODLE

**Л.М. Заміховський**

Доктор технічних наук, професор, проректор з наукової роботи\*

**Л.М. Ходек**

Аспірант\*

\*Галицька Академія

вул. Вовчинецька, 227, м. Івано-Франківськ, 76006

E-mail: ime@ime.if.ua, sundew2006@rambler.ru

*В роботі проаналізовані причини використання однопараметричної моделі Раша та розглянуто побудову компонентів специфічного математичного двигуна для інтеграції її в систему дистанційного навчання Moodle*

*В работе проанализированы причины использования однопараметрической модели Раша и рассмотрено построение компонентов специфического математического двигателя для интеграции ее в систему дистанционной учебы Moodle*

*This article presents causes of usage of Rasch model and process of building specific math engine and it's integration into LMS Moodle*

З появою інформаційних технологій все більшої популярності набуває застосування елементів дистанційного навчання при наданні освітніх послуг. Одна з ключових проблем впровадження дистанційної форми навчання (ДН) – проведення процедури діагностування рівня знань студентів. Основним інструментом діагностування рівня знань в ДН є використання тестів. При цьому слід зауважити, що:

- розробка тестів, здатних адекватно оцінити рівень знань студентів, – це не просто складання завдань і об'єднання їх в тест. Тест – це система завдань, в якій кожне завдання повинне задовольняти певним критеріям;

- не можна зводити перевірку знань до одного лише тестування. За допомогою тестів не можна перевірити, наприклад, доказ теорем, і, звичайно ж, ніяке тестування не замінить спілкування студента з викладачем.

Тому зазвичай при традиційному методі навчання тестування застосовується як перший етап складання іспиту, за результатами якого здійснюється допуск до другого етапу – співбесіди з викладачем [1].

Розробка тестових завдань та обробка отриманих результатів є дуже важливим етапом впровадження ДН, однак не менш важливе значення при цьому має послідовність представлення тестових завдань студентові та метод визначення рівня знань за результатами тестування (модель тестування). При вимірюванні рівня знань згідно класичної моделі тестів, для кожного студента формується тест певної довжини з вибраних випадковим чином тестових завдань.

При цьому тести відрізняються за складністю, а результуюча оцінка формується за кількістю вірних відповідей з урахуванням вагових коефіцієнтів завдань.

Недоліком такого підходу є поява ситуацій, при яких слабкому студенту попадається сильний тест і, як наслідок, – практична відсутність відповідей і навпаки, сильному студенту попадається легкий тест і він не може реалізувати своїх можливостей.

Для отримання максимальної інформативності результатів діагностування необхідно, щоб середня складність тесту відповідала його гіпотетичному рівню знань.

Такі проблеми вирішуються за допомогою адаптивного тестування, яке представляє собою комп'ютеризовану систему науково-обґрунтованих перевірок та оцінки результатів навчання, яка має досить високу ефективність за рахунок оптимізації процедур генерації, пред'явлення та оцінки результатів виконання адаптивних тестів і дозволяє отримати на виході деякий бал за тест (первинний бал) [2].

Одним з варіантів вирішення даної проблеми є застосування теорії тестових завдань (Item Response Theory - IRT), зокрема однопараметричної теорії Раша, яка по суті є основою теорії IRT [3]. Вибір даної теорії був зумовлений наступними чинниками:

- Відносна простота реалізації, що дозволяє застосовувати цю теорію в межах будь-якої системи ДН. Незважаючи на цей фактор, дана модель не є реалізованою для сучасних програмних оболонок, на основі яких будуються дистанційні курси, наприклад Moodle.

- Стійкість розрахованих значень рівня знань і складності завдання, їх відносна інваріантність. Це дозволяє стверджувати, що подібні моделі краще, ніж будь-які інші дозволяють оцінити риси окремої особистості, які недоступні безпосередньому вимірюванню. Стійкість вимагає наявності значної кількості студентів, що підлягають тестуванню.

Оцінка складності тестових завдань не залежить від вибірки випробовуваних (студентів), на яких була отримана, і оцінка рівня знань студентів, відповідно, не залежить від набору тестових завдань, що був використаний.

Пропуск даних для деяких комбінацій (тестуючий - тестове завдання) не є критичним.

Не зупиняючись на розгляді особливостей однопараметричної моделі Раша, яка є достатньо відомою, більш детально зупинимося на процедурі її впровадження в навчальний процес з використанням оболонки Moodle.

Модель Раша спирається на такі поняття, як «складність завдання» і «рівень знань». Так, одне завдання вважається за складніше, ніж інше, якщо вірогідність правильної відповіді на перше завдання менша, ніж на друге, незалежно від того, хто їх виконує.

Аналогічно, більш підготовлений студент має велику вірогідність правильно відповісти на всі завдання, чим менш підготовлений [3].

Завдяки простій структурі моделі, існують зручні обчислювальні процедури для перевірки адекватності моделі: для всього набору тестових результатів, для кожного студента, що тестується, для кожного завдання та для кожної конкретної відповіді.

Основною математичною функцією теорії Раша є однопараметрична логістична функція Раша, яка називається характеристичною кривою [4]

$$P(S,t) = \frac{S}{S+t} = \frac{1}{1+\frac{t}{S}} \quad (1)$$

Вірогідність  $P(S,t)$  отримала назву функції успіху. Як видно з виразу (1), функція успіху залежить лише від відношення  $t$  до  $S$ , тому модель Раша називається однопараметричною і використовує шкалу стосунків.

Вводячи нові змінні:

$$\ln S = \theta, S = \exp(\theta), \ln t = \delta, t = \exp(\delta) \quad (2)$$

Вираз (1) можна переписати у вигляді:

$$P(\delta, \theta) = \frac{1}{1 + \exp(\delta - \theta)} \quad (3)$$

Формула (3) є основним рівнянням однопараметричної логістичної моделі Раша, одиниця вимірювання  $\delta$  і  $\theta$  називається логітом.

При одному логіті  $\delta_0 (=1 \text{ і } \theta_0 =1)$  вірогідність успіху  $P(\delta, \theta) = 0,5$ , тобто вірогідність виконання стандартного завдання стандартним учасником має дорівнювати 0,5 (рис. 1).

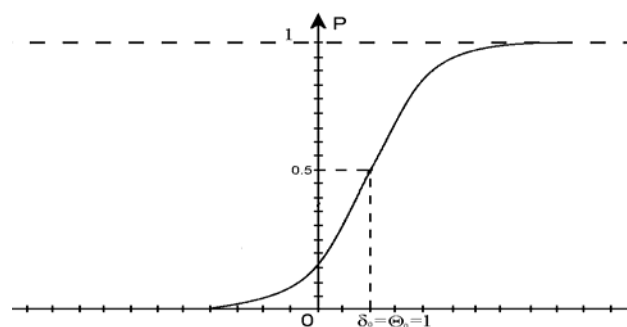


Рис. 1. Функція успіху однопараметричної моделі Раша

Основна проблема, яка пов'язана з однопараметричною моделлю Раша, полягає в інтеграції можливостей моделі з навчальним процесом, який є встановленим з допомогою оболонок дистанційного навчання, таких як Moodle.

Це означає розробку уніфікований інтерфейс, який можна було б застосувати в навчальному процесі,

як частину обробки результатів адаптивного тестування.

Було розроблено об'єктно-орієнтовну модель, яка дозволяє застосовувати двохпараметричну модель Раша в комплексі навчального процесу, як додатковий модуль Moodle (рис. 2). Для цього нам слід здійснити декілька кроків:

Розробити уніфікований формат даних, в якому ми будемо зберігати дані моделі.

2. Створити математичний двигун для обробки даних та представлення результатів у вигляді спеціальних об'єктів.

3. Створити модуль для графічного відображення результатів.

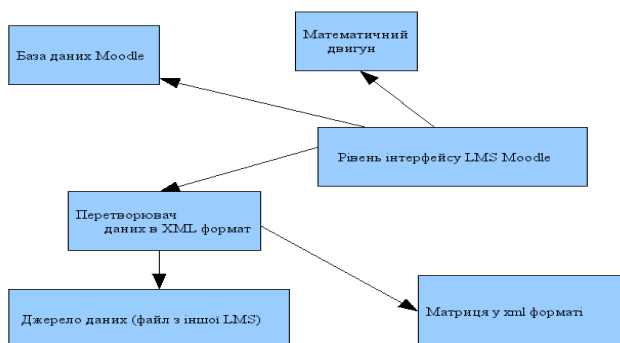


Рис. 2. Структурна схема інтеграції модулів для побудови функції успіху

Дані, що використовуються для обробки математичним двигуном, можуть поступати з двох джерел (рис. 2): безпосередньо з бази даних Moodle і у вигляді файлу даних, що був експортований іншою системою управління навчанням (LMS) та приведенний до відповідного xml формату.

Файл обробляється спеціальною програмою (парсер), що представляє дані у вигляді об'єктів матриці, яка потім подається на вхід математичного двигуна, що повертає результати у вигляді функції успіху.

Завдання інтерфейсу – організувати взаємодію між користувачем та математичним двигуном, передаючи дані з рівня користувача до математичного двигуна і навпаки.

Для зберігання даних, була створена власна структура xml формату

```
<matrix group="CU-06-3" course="ONTDS">
  <user name="Амброзьяк">
    <question>0</question>
    <question>1</question>
    .....
  </user>
  <user name="Голімбловська">
    <question>1</question>
    <question>1</question>
    .....
  </matrix>
```

Особливістю даної структури є те, що можна згенерувати матрицю, яка характеризує будь-яку вибірку і передати її на вхід математичного двигуна.

Наступний крок – це обробка даних та побудова функції успіху.

Наведемо реалізацію модуля Раша з використанням скриптової мови програмування PHP, який обчислює функцію для двохпараметричної моделі Раша.

```
<?php
class Rasch {
/**
 * Функція models_get_q() реалізує алгоритм
 * однопараметричної моделі Раша, що повертає
 * латентні параметри рівня підготовленості
 * тестуючого до даного тесту
 * Всі обчислення проводяться з точністю
 * до трьох знаків після коми
 *
 * @return double[] $Qstatochni одновимірний
 * масив дійсних значень
 */
public function models_get_q() {
    $m = new Matrix();
    // присвоюємо змінним отримані значення
    // тестування з класу Matrix
    $matrix = $m->printbal;
    $students = $m->amount_users;
    $question = $m->amount_question;

    // обчислюємо первинні бали по
    // студентам $b та питанням $c
    $b = $this->models_get_primary_bal_students ( $matrix, $students,
    $question );
    $c = $this->models_get_primary_bal_question ( $matrix, $students,
    $question );

    // Підраховуємо долю правильних рі відповідей
    // кожного тестуючого на завдання тесту
    $pi = $this->models_get_pi ( $b, $students, $question );

    // Підраховуємо по всій групі тестуючих долю вірних
    // рі відповідей на кожне завдання
    $pj = $this->models_get_pj ( $c, $students, $question );

    // Проводимо попередню оцінку значень параметра q та d
    $q = $this->models_get_poperednya_ocinka_pidgotovku ( $pi );
    $d = $this->models_get_poperednya_ocinka_skladnosti ( $pj );

    // Знаходимо середнє значення попередньої оцінки q та d
    $qser = $this->models_get_serenya_ocinka ( $q );
    $dser = $this->models_get_serenya_ocinka ( $d );

    // Переносимо на шкалу з єдиним початком відліку
    // попередньої оцінки q
    $sqmichene = $this->models_get_latentni_parametru_na
    _shkali_vidliky ( $q, $qser );

    // Обчислюємо оцінку дисперсії для q та d
    $duspersiyaq = $this->models_get_ocinka_duspersi ( $q, $qser );
    $duspersiyad = $this->models_get_ocinka_duspersi ( $d, $dser );

    // Визначаємо коректуючі множники qlambda
    $qlambda = $this->models_get_lambda_q ( $duspersiyaq, $duspersiyad );

    // Вирівнюємо масштаби шкал, використовуючи коректуючий множник
    $qstatochni = $this->models_get_korekciya_shkalu ( $sqmichene, $qlambda );
    return $qstatochni;
}
?>
```

Даний модуль, який по суті представляє собою математичний двигун, що обчислює оціночку функції, може бути використаний у будь-якій оболонці дистанційного навчання.

Маючи двигун, який рахує значення для функції успіху, необхідно інтегрувати його в систему дистанційного навчання Moodle та побудувати веб-інтерфейс.

Так як код, що використовується для побудови інтерфейсу є дуже великим, то обмежимося кодом, який використовується для побудови форми даних у вигляді графіка

```
$object = new DAO ();
    $object->form ();
    $object->initialization ();
    $object->get_matrix ();
    $object->get_rasch();
    $students = 5;
    $question = 10;
    for ($i = 0; $i < $students; $i ++){
    for ($j = 0; $j < $question; $j ++){
    echo " | ".$object->printbal[$i][$j];
    }
    }
    echo "<br>";
```

---

### Висновок

---

В результаті проведеної роботи, вперше було розроблено структуру модуля до LMS Moodle, який дозволяє побудувати функцію успіху для однопараметричної моделі Раша. Використання модуля в системі

управління навчанням, зокрема при проведенні тестування рівня знань студентів дозволяє на основі отриманих статистичних даних визначити імовірність вірної відповіді з врахуванням рівня знань студента і провести правильне ранжування рівня тестових питань і підготовки тестових завдань для конкретної групи чи потоку студентів відповідних спеціальностей.

---

### Література

1. <http://www.wikiznanie.ru/>
2. [www.testor.ru](http://www.testor.ru)
3. Аванесов В.С. Композиция тестовых заданий. – М.: Из-во Центра тестирования Минобразования РФ, 2002, - 239С.
4. Переверзев В.Ю. Критериально - ориентированные педагогические тесты для итоговой аттестации студентов. М.: Из-во НМЦ СПО Минобразования РФ, 1998, -152 С.